

Letter

SNS-CF: Siamese Network with Spatially Semantic Correlation Features for Object Tracking

Thierry Ntwari , Hasil Park , Joongchol Shin  and Joonki Paik * 

Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul 06974, Korea; jonthierry@ipis.cau.ac.kr (T.N.); hspark@ipis.cau.ac.kr (H.P.); jcshin@ipis.cau.ac.kr (J.S.)

* Correspondence: paikj@cau.ac.kr

Received: 10 June 2020; Accepted: 25 August 2020; Published: 28 August 2020



Abstract: Recent advances in object tracking based on deep Siamese networks shifted the attention away from correlation filters. However, the Siamese network alone does not have as high accuracy as state-of-the-art correlation filter-based trackers, whereas correlation filter-based trackers alone have a frame update problem. In this paper, we present a Siamese network with spatially semantic correlation features (SNS-CF) for accurate, robust object tracking. To deal with various types of features spread in many regions of the input image frame, the proposed SNS-CF consists of—(1) a Siamese feature extractor, (2) a spatially semantic feature extractor, and (3) an adaptive correlation filter. To the best of authors knowledge, the proposed SNS-CF is the first attempt to fuse the Siamese network and the correlation filter to provide high frame rate, real-time visual tracking with a favorable tracking performance to the state-of-the-art methods in multiple benchmarks.

Keywords: object tracking; siamese network; correlation filter; spatially semantic correlation features

1. Introduction

Visual object tracking aims at estimating the position of an arbitrary target in a video sequence by establishing a correspondence between similar pixels of different frames [1–3]. It finds a wide range of usage in intelligent video analysis applications such as automatic visual surveillance, autonomous driving, augmented reality, and action recognition tasks, to name a few.

Despite the tremendous progress of visual tracking over the past few years, we still face the rise of numerous challenges including fast motion, illumination variation, occlusion, background clutter, intraclass variations, and so forth.

To alleviate the above challenges, we will learn better and more robust features that improve visual object tracking algorithms [4]. We adopted the same idea to deep learning algorithms using the most important features in the network. Another remedy for the above challenges is that object tracking changed its gears to an alternative approach in which, a Deep CNN is trained to address a more general similarity learning (Siamese learning) problem in an initial offline phase, and then this function is simply evaluated online during tracking, as explained in Bertinetto and Luca et al. [5].

Thanks to the findings that the straightforward replacement of shallow backbone with deeper and wider networks does not bring much improvement to Siamese network, the notorious accuracy gap to Siamese network counterparts is remarkable as described in References [1,2,6] but still, Reference [3,7] proved that spatially semantic correlation features are necessary to boost even further the accuracy gap.

The most challenging part of visual tracking is the real-time or online tracking as shown in Figure 1, where the tracker cannot use future frames to infer the current position of an object [8].

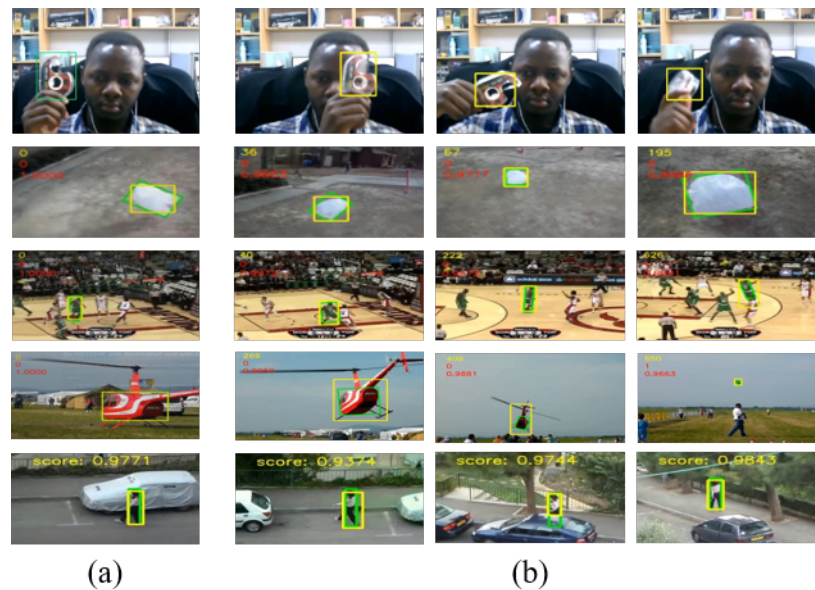


Figure 1. Illustration of the visual object tracking results: (a) The initial frame with a bounding box. (b) Tracking results in three different selected frames with the ground truth in green and ours in yellow bounding boxes. From top to bottom: Scotch tape webcam, Bag, Basketball, Helicopter and Woman, respectively. Apart from Scotch tape webcam, videos are from VOT2018 [9] and OTB2015 [10] datasets.

In this work, we address the accuracy gap and frame update problems of the Siamese network and correlation filters, respectively, in a twofold contribution:

- We extract spatially semantic correlation features (SSF) from the Siamese network.
- We learn adaptive correlation filters (ACF) at every convolutional layer output and calculate their weighted sum in the end.

In the reminder of this paper, we briefly review related works in Section 2, followed by the proposed method in Section 3. In Sections 4 and 5, we implement and evaluate our method. Finally, we conclude the paper in Section 6.

2. Related Works

In this section, we briefly describe deep Siamese tracking and correlation filters in Sections 2.1 and 2.2, respectively.

2.1. Deep Siamese Tracking

Zhang et al. [1] and Li et al. [2] have recently proved that the Siamese network can benefit from deeper backbone networks using end-to-end learning. Based on those works, Siamese networks formulate object tracking as a cross-correlation problem between two input signals, one of which is an interested region of an image, and the other is a relatively larger search window in another image [1,2,5,6]. Training the Siamese network involves a Y-shaped network that joins two branches, one of which predicts the object template (interested image), and the other predicts the search region (search window). This process consists of two steps—(1) an offline training [1,5] for a similarity function learning between the two input signals by cross-correlating them, and (2) an online training for the similarity function update as the tracking goes on [5]. With the addition of spatially semantic correlation features (SSF) and adaptive correlation filters (ACF), we improved both accuracy and speed of the deep Siamese networks.

2.2. Correlation Filter Tracking

Correlation filters have attracted attention in the tracking field during the last decade due to their high computational efficiency in the Fourier domain and the kernel trick method [11,12]. This consists of a form of circular shifts of input signals to a target Gaussian function which does not require hand-crafted features of the target. Correlation filter related works, HOG or color-attributes presented a frame update problem and used hand-crafted features [12]. Therefore, we address these by finding multiple correlation filters in hierarchical convolutional layers as opposed to only one single filter at the classification/regression layer of the network used by existing approaches.

3. Proposed Method

This section describes the proposed algorithm as shown in Figure 2 and revisits the twofold contributions, as mentioned in the previous section. It will as well explain preliminaries to understand the proposed contributions.

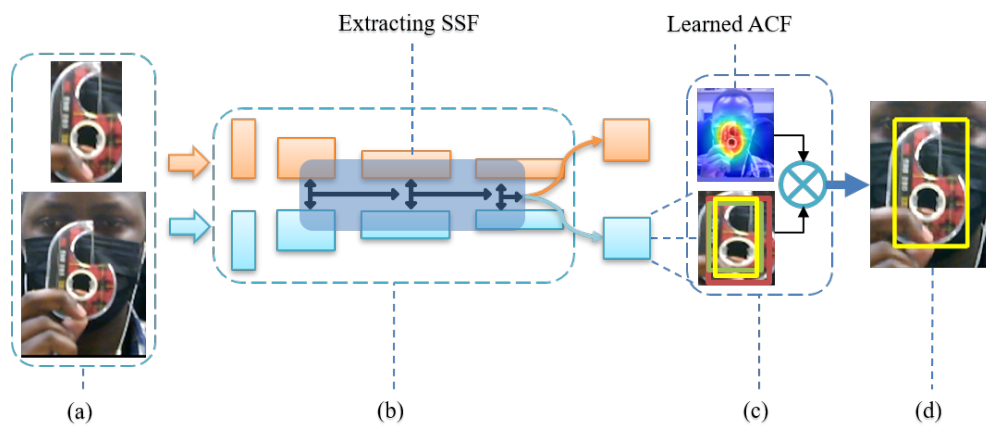


Figure 2. Spatially semantic correlation features (SNS-CF) proposed algorithm: (a) A pair of input images and the corresponding search window, (b) the Siamese network with three region proposal networks in a shadowed middle part, which highlights spatially semantic correlation features of interest, (c) the correlation search module, with multiple up-sampled bounding boxes convolved with a learned adaptive correlation filter (ACF), and (d) output of the predicted final tracking result with a yellow bounding box.

3.1. Siamese Net

Bertinetto et al. first proposed the Siamese network, called SiamFC [5], and Li et al. improved it by using region proposal networks [6]. Recently, Li et al. made further improvements by solving the problem of a small receptive field, network stride and padding while reducing the translated image z and a candidate search image x . The image z represents the object of interest, while x is typically larger and represents the search area in subsequent video frames. Both inputs are processed by a ConvNet φ with parameters θ . This yields two feature maps, which undergo a cross-correlation:

$$f_{\theta}(z, x) = \varphi_{\theta}(z) * \varphi_{\theta}(x) + b_{-1}, \quad (1)$$

where b_{-1} denotes a positive offset to model the similarity value. This ensures the efficient training and inference by obeying intrinsic restrictions for structure symmetry, that is, $f(z, x') = f(x', z)$, which is appropriate for the similarity learning. Equation (1) performs an exhaustive search of the pattern z over the image x to match the maximum value in the response map f to the target location. This is done through an offline training with random image pairs x, z taken from training videos and the corresponding ground truth label y . The parameters θ of the ConvNet φ are obtained by minimizing the logistic loss \mathcal{L} over the training set:

$$\arg \min_{\theta} \mathbb{E}_{(z,x,y)} \mathcal{L}(y, f_{\theta}(z, x)). \quad (2)$$

3.2. Region Proposal Network (RPN)

The Siamese network weights the similarity measure between the input image and the search window. We need an extra fragment installed in adjacent layers of the network, and the choice of where and how many is a hyperparameter. This extra fragment is used to refine the proposal. It consists of a pair-wise correlation section with two branches as well, one for classification of background and/or foreground, and another for regression of proposal. More about these RPNs are found in a pioneering work by Li et al. [6]. We made three RPNs and implanted them in our modified ResNet50 [13] to capture spatially semantic information. RPN1, RPN2 and RPN3 aggregate multi-branch features of conv 3 (res3d_branch2c), conv4 (res4f_branch2c) and conv5 (res5c_branch2c), respectively. The extraction of such information used in tracking tasks follows in the next section.

3.3. Extracting SSF

We aggregate different deep layers into RPNs following Reference [6]. The three RPNs are located on the richest middle layers, as shown in Figure 3.

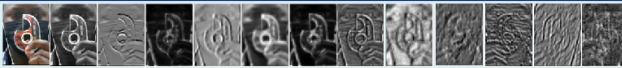
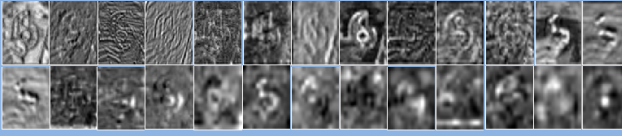
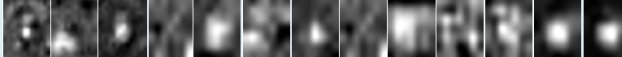
Layers	Specs	Graphical view
Earlier Layers	Spatial info. Low Robustness	
Middle Layers	Tracking info. Spatially semantic correlation features: Accuracy – Robustness balance	
Deeper Layers	Semantic info. Low Accuracy	

Figure 3. Graphical view of spatially semantic correlation features. From Left to right, top to bottom we have a sequence of activations from a deep convolutional neural network (CNN) (ResNet50 [13] for our case), where the first (top) row consists of Earlier layer activations poor in robustness, and the last (bottom) row consists of deeper layer activations poor in accuracy. To balance the robustness—accuracy trade-off, our method suggests to use middle layer activations (2 middle rows) rich in both spatial and semantic information.

The idea of extracting SSF comes from the need to improve existing features. Dimitris et al. significantly improved classification features by applying robust optimization techniques [4]. On the other hand, Erhan et al. decided on good features to correlate for visual tracking by utilizing robust features that are invariant to any kind of appearance change of the object, while predicting the object location as properly as in the case of no appearance change [14]. Other approaches used hierarchical features [3], spatially semantic features [7] and hierarchical attention weights [15] to define appropriate features for object tracking in CNNs.

In our task, we use aggregated layers in RPNs to collaboratively infer the target localization. As for ResNet50 [13], we explore multi-level features extracted from the last three aggregated layers. We refer to these outputs RPN1, RPN2 and RPN3 as x_3 , x_4 and x_5 , respectively. They constitute both scores S and bounding boxes B and as we mentioned before, we drop scores and use only bounding boxes. We will perform an interpolation of B as shown in Figure 2, to have the same spatial resolution, (see Section 3.6) to be able to perform a correlation search. At each RPN we perform a weighted sum directly as they have same individual spatial resolution, and a weighted-fusion layer combines all the outputs as:

$$\mathcal{B}_{all} = \sum_{l=3}^5 \beta_l * \mathcal{B}_l, \quad (3)$$

where \mathcal{B}_{all} denotes the bounding boxes on weighted-fusion layer, \mathcal{B}_l denotes the bounding boxes on l th layer and β_l denotes the interpolation factor.

3.4. Convolutional Features

Extraction of convolutional feature maps encodes target appearances. The forward propagation along the network strengthens semantic discrimination, while the spatial information gradually reduces. As shown in Figure 1, it is easy to locate the scotch tape in earlier layer activation maps, but it gets blurry in deeper network layers. Since only middle layers conserve spatially semantic information, we ignore both earlier and deeper layers, and put our focus on middle layers [7,15]. Conventionally, CNNs use different operators, pooling being one of them, which result in shrinking the spatial resolution with the increase in the depth of convolutional layers. For instance, the size of res5a_branch2b, the 145th convolution of ResNet50 [13] is $7 \times 7 \times 512$ which is $\frac{1}{32}$ of the input size of $224 \times 224 \times 3$. To preserve the spatial resolution, we bilinearly interpolate each feature map to a fixed size as:

$$x_i = \sum_k \alpha_{ik} h_k, \quad (4)$$

where α_{ik} denotes the interpolation weight and i, k denote the position of neighboring feature vectors, respectively. More details on connecting features from multiple layers are found in Reference [16] for segmentation and fine-grained localization using CNNs.

3.5. Correlation Filters

Typical correlation filters [7,17,18] learn a discriminative classifier and estimate the translation of target objects by searching for the maximum correlation response. Correlation filters have been very competitive, thanks to working in the Fourier domain, where circular shifts are computed in a lapse of time using kernel trick [11,12]. The circular shifts are defined as:

$$x = \{x_{m,n} | (m,n) \in \{0,1,\dots,M-1\} \times \{0,1,\dots,N-1\}\}, \quad (5)$$

where x denotes the l th layer of feature vector of size $M \times N \times D$. M , N and D denote width, height, and number of channels, respectively. x^l concisely denotes x on the layer l , implicitly with its dependencies, M , N and D . $M-1$ and $N-1$ denote the circular forms of x in both directions. The common characteristic of circular shifts is their Gaussian function label $y(m,n)$, determined as:

$$y(m,n) = e^{-\frac{(m-M/2)^2 + (n-N/2)^2}{2\sigma^2}}, \quad (6)$$

where σ denotes the kernel width. A correlation filter w with the same size of x is then learned by solving the following minimization problem:

$$w^* = \arg \min_w \sum_{(m,n)} ||w \cdot x_{m,n} - y||^2 + \lambda ||w||_2^2, \quad (7)$$

where λ denotes a positive regularization parameter, and the inner product is induced by a linear kernel in Hilbert space [3]. The core ingredients in CNNs are the ability to learn by training and avoiding handcrafted samples. Therefore, the correlation filter in the Fourier domain described in (7) can save a tremendous amount of time by solving it in each individual feature channel using the fast Fourier transform (FFT). Capital letters denote the corresponding small letter signals in Fourier transformed signals. The learned filter in the frequency domain on the d th ($d \in \{1, \dots, D\}$) channel can be written as:

$$W^d = \frac{Y \odot \bar{X}^d}{\sum_{i=1}^D X^i \odot \bar{X}^i + \lambda}, \quad (8)$$

where Y is the Fourier transform of $y = \{y_{m,n} | (m,n) \in \{0,1,\dots,M-1\} \times \{0,1,\dots,N-1\}\}$, following (5), and \odot denotes the Hadamard (element-wise multiplication) product operator.

3.6. Learned ACF

Also known as the maximum of the correlation response map, given an image patch in the next frame, the feature vector on the l th layer is denoted as z of size $M \times N \times D$. The l th correlation response map is computed as:

$$f_l = \mathcal{F}^{-1} \left(\sum_{d=1}^D W^d \odot \bar{z}^d \right), \quad (9)$$

where \mathcal{F}^{-1} denotes the inverse FFT operation. The learning of ACF is completed in searching for the position of the maximum value of Equation (9) with the same size. It is cross-correlated with interpolated bounding boxes (\mathcal{B}) in Section 3.3, to find the optimized target location.

4. Implementation Details

SNS-CF algorithm is a modified ResNet50 [13] to perform proposal classification and bounding box regression. We added three 1×1 randomly initialized convolutional layers to conv3, conv4, and conv5 to reduce the feature dimension down to 256. During training, it is optimized using Stochastic Gradient Descent (SGD) method, which can benefit from parallel computing using 8 GPUs with a total of 128 pairs per minibatch, that is, 16 pairs per GPU, to reduce a week of training into just 12 h. We initially used a single GPU with 16 pairs, initial learning rate of 0.001 for first 5 epochs to train RPN branches. The entire network is trained in an end-to-end manner, and in the end, 15 last epochs are trained with an exponential learning rate decay from 0.004 to 0.0004, with a momentum of 0.9. The training loss is the sum of standard smooth loss \mathcal{L} in (2) and the correlation filter loss w^* in (7).

5. Experimental Results

Hardware specifications—SNS-CF algorithm is implemented using Python [19] and evaluated on MATLAB (Natick, MA, USA) [20] LaSOT evaluation toolkit [21], Intel [22] i7-8700K 3.70 GHz CPU with 32 Mb RAM and a single NVIDIA (Santa Clara, CA, USA) [23] GeForce GTX 1080 Ti. Dataset—SNS-CF algorithm is evaluated on widely used tracking datasets, for instance OTB-2015 [10], VOT-2018 [9], and LaSOT [21]. OTB-2015 [10] consists of 100 video sequences, VOT-2018 public dataset [9], one of the most recent datasets for evaluating online model-free single object trackers consists of 60 video sequences while LaSOT [21] dataset provides a large-scale, high-quality dense annotations with 1400 videos in total and 280 videos in the training set.

Metrics—OTB-2015 [10] is evaluated following the evaluation protocol in Reference [10], and has three following metrics, Distance Precision rate (DP), Overlap ratio (OS), and Center Location Errors (CLE). VOT-2018 [9] is evaluated following the evaluation protocol in Reference [9]. We adopt the Expected Average Overlap (EAO), Accuracy (A), Robustness (R), and no-reset-based Average Overlap (AO) to compare different trackers. Lastly, LaSOT [21] is evaluated following evaluation protocol in Reference [21] with Distance Precision (DP) and Overlap Success (OS) plots over 100 benchmark sequences using One-pass evaluation (OPE) on both threshold and Area Under the Curve (AUC). We will present the evaluation results with respect to each dataset shortly.

Training—The backbone network of SNS-CF algorithm is ResNet50 [13] pre-trained on ImageNet [24] for image labeling, as a good initialization to other tasks, even though it is quite old now. In both training and testing, we followed SiamFC [5] protocol and used an exemplar and search images patches of 127×127 and 255×255 pixels respectively. We randomly translated up to ± 8 pixels and re-scaled $2^{\pm 1/8}$ and $2^{\pm 1/4}$ for exemplar and search images, respectively. We trained our network on the training sets of Imagenet-VID [24], COCO [25], and Youtube-VOS [26].

Evaluation method—We perform the evaluation of our algorithm with respect to correlation filter—based trackers, and Siamese network—based trackers. We will conduct separate evaluation and provide results for each category. Starting from correlation filter—based trackers, we quantitatively evaluated the proposed algorithm with 9 state-of-the-art trackers [3,12,27–33], considering the distance precision rate (DP) at 20 pixels, overlap success rate (OS) at 0.5, center location errors (CLE) and tracking speed, from 100 sequences of OTB-2015 [10] benchmark.

Second, the proposed algorithm is evaluated compared to Siamese networks—based trackers, and we will focus on the short-term single object tracking on OTB2015 [10] and VOT2018 [9], and analyze the generalization of our method on LaSOT [21], the most recent largest benchmark for single object tracking. Short-time single object tracking, as opposed to long-term single object tracking is the scenario where the object has to stay in the field of view throughout the tracking, or just for a fraction of time leaves the field of view or becomes fully occluded.

CF-based results—We present results from evaluating the proposed algorithm with respect to 8 correlation filter—based state-of-the-art trackers [3,12,27–33]. They can be broadly categorized into three classes that is, deep learning trackers (DL-Trackers) [27], correlation filter trackers (CF-Trackers) [12,28,34] and representative online classifier trackers (ROC-Trackers) [29,31–33]. Table 1 illustrates the quantitative comparisons of distance precision rate (DP) at 20 pixels, overlap success rate (OS) at 0.5, center location errors (CLE), and tracking speed, from 100 sequences of OTB-2015 [10] benchmark. It shows a favorable performance against the state-of-the-arts.

Table 1. Quantitative evaluation of SNS-CF algorithm with eight state-of-the-art trackers [3,12,27–33], considering the distance precision rate (DP) at 20 pixels, overlap success rate (OS) at 0.5, center location errors (CLE) and tracking speed, from 100 sequences of OTB-2015 [10] benchmark. Red and blue numbers indicate the best and second best results, respectively.

Metrics	Ours (SNS-CF)	CF2 [3]	KCF [12]	Struck [29]	DLT [27]	STC [28]	TLD [33]	MIL [32]	CT [31]
DP rate ↑ (%)	84.0	83.7	69.2	63.5	52.6	50.7	59.2	43.9	35.9
OS rate ↑	65.7	65.5	54.8	51.6	43.0	31.4	49.7	33.1	27.8
CLE ↓ (pixel)	20.2	22.8	45.0	47.1	66.5	86.2	60.0	72.1	80.1
Speed ↑ (FPS)	35.1	10.4	243	9.84	8.43	653	23.3	28.0	44.4

Table 1 shows the highest tracking result against state-of-the-art trackers in terms of DP, OS and CLE, which are roughly comparable to Reference [3]. KCF [12], second fastest and STC [28], the fastest use handcrafted features, which do not require high computational complexity and time as deep CNN feature—based do. However, our tracker runs at an average of 35.1 fps, which is fairly good among CNN—based trackers.

Siamese-based results—We present results from evaluating the proposed algorithm with respect to VOT-2018 [9] and LaSOT [21] benchmarks. First, we start from VOT-2018 [9] and test our tracker SNS-CF against 7 state-of-the-art methods containing either correlation filters or Siamese networks or both [2,6,35–42]. We follow its evaluation protocol and present results in the following Table 2.

Table 2. Comparison with the state-of-the-art trackers in terms of Expected Average Overlap (EAO), Robustness and Accuracy on the VOT-2018 [9]. Red and blue numbers indicate the best and second best results, respectively.

Metrics	Ours (SNS-CF)	SiamRPN++ [2]	LADCF [38]	MFT [39]	SiamRPN [6]	UPDT [40]	SA_Siam_R [41]	DRT [42]
EAO ↑	0.423	0.414	0.389	0.385	0.383	0.378	0.337	0.356
Accuracy ↑	0.587	0.600	0.503	0.505	0.586	0.536	0.566	0.519
Robustness ↓	0.223	0.234	0.159	0.140	0.276	0.184	0.258	0.201
AO ↑	0.487	0.498	0.421	0.393	0.472	0.454	0.429	0.426

Takeaways from Table 2 are interesting as we can notice that the proposed algorithm achieves the best Expected Average Overlap rate (EAO) against all the state-of-the-arts, with a gain of roughly 1% to the baseline and top performing. The accuracy is about 1.3% short of the baseline, but also higher than any other state-of-the-art. The robustness is 1.1% higher than the baseline, but unfortunately still lower than the VOT-2018 [9] challenge winner MFT [39], mostly because the latter is armed with Multi-hierarchical independent correlation filters, a close technology to our algorithm. Notice that we outperform it in the rest of the metrics. Lastly, the overall One Pass Evaluation (OPE) is also adopted to evaluate trackers and the AO values are reported to demonstrate their performance. Our algorithm achieved second best value to the baseline and overall benchmark.

Second, we further validate the proposed algorithm by testing it on a larger and more challenging dataset, LaSOT [21]. We follow its evaluation protocol and report the overall performances in Figure 4.

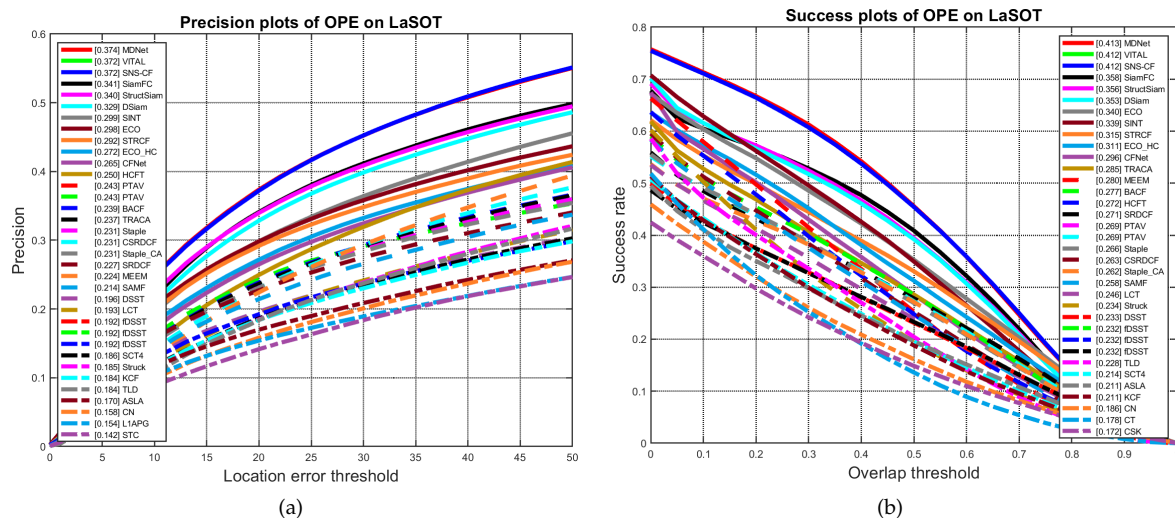


Figure 4. (a) Distance Precision and Overlap Success plots over 100 benchmark sequences using One-pass Evaluation (OPE) on both threshold scores at 20 pixels and (b) Area Under The Curve score (AUC) (right) on LaSOT [21]. Notice that the proposed algorithm (Blue) ranks third, with first three trackers MDNet [43], VITAL [44] and SNS-CF (ours) merely having the same performance. Extensive assessments over fifteen challenging tracking scenarios are experimented and results are available from authors upon request. We reproduced Figure 4 using MATLAB official LaSOT Evaluation toolkit [21].

Fusion-based results—We present results from combining state-of-the-arts of both correlation filter-based tracker [3] and Siamese network-based tracker [2] with direct combination, that is, with no modification, and with our proposed algorithm that includes the extraction of SSF and the learning of ACF. The following Table 3 has the details.

Table 3. Comparison with correlation filter, Siamese network, and the proposed SNS-FC algorithm that fuses both technologies. We present the results on VOT-2018 [9]. Red and blue numbers indicate the best and second best results, respectively.

Metrics	Ours (SNS-CF)	Correlation Filter [3] (Baseline)	Siamese Network [2] (Baseline)	Both (Without SSF and ACF)
DP rate \uparrow (%)	84.0	83.7	-	83.3
OS rate \uparrow (%)	65.7	65.5	-	65.0
CLE \downarrow (pixel)	20.2	22.8	-	21.3
EAO \uparrow	0.423	-	0.414	0.420
Accuracy \uparrow	0.587	-	6.00	0.557
Robustness \downarrow	0.223	-	0.234	0.228
Speed \uparrow (FPS)	35.1	10.4	-	34.2
AO \uparrow	0.487	-	0.489	0.393

Table 3 on the preceding page shows that SNS-CF algorithm clearly improves both the correlation filter and Siamese network trackers in a number of metrics. The last column indicates the direct combination of correlation filter tracker [3] and Siamese network [2] without our contributions, and we remark an early improvement in CLE, EAO, robustness and speed, thanks to both the advantages of deep CNN features as opposed to handcrafted HOG features, and the Fourier domain of correlation filters that dramatically improves the speed [11,12]. The first column shows that the proposed algorithm outperforms both correlation filter and Siamese network baselines in general, thanks to spatially semantic (SSF) features and the learning of adaptive correlation features (ACF).

Table 4 shows that SNS-CF performs best on both intra-class and illumination variations, while it is the second best on occlusions due to the lack of a re-detection module. On the whole, the proposed SNS-CF shows a significant improvement in robustness.

Table 4. Comparison of robustness on different SOT challenging problems with correlation filter, Siamese network, and the proposed SNS-FC algorithm. We present the results on VOT-2018 [9]. Red and blue numbers indicate the best and second best results, respectively.

Scenarios	Ours (SNS-CF)	Correlation Filter [3] (Baseline)	Siamese Network [2] (Baseline)
Background clutter	0.228	0.225	0.230
Intra-class variations	0.198	0.302	0.227
Occlusions	0.312	0.226	0.232
Illumination variations	0.154	0.159	0.247
Average Robustness ↓	0.223	0.228	0.234

Failure cases—In some challenging scenarios, our algorithm failed completely to locate to position of the target. We suspect this is due to intense background clutter, appearance of many similar foreground images, although not targets, and severe out-of-view. Some other cases include bright background and dark foreground, where the first layer features are enough to check failure instead of using all the SSF features. Severe out-of-view cases may be well addressed if our algorithm was equipped with a re-detection module, which will be our future research. This is illustrated in Figure 5, whereas correctly located targets are illustrated in Figure 1.



Figure 5. Failure cases. Video instances of Basketball, Helicopter and Woman on VOT-2018 [9] and OTB2015 [10]. They represent multiple foreground images similar to the target, severe out-of-view and sudden background clutter respectively.

6. Conclusions

In this paper, we proposed a novel effective fusion algorithm called SNS-CF, which trains a Siamese network and a correlation filter for visual object tracking. We used the fading correlation filter technology to improve the popular Siamese network. The similarity search technique of a typical Siamese network, fused with correlation filter, alongside spatially semantic correlation features from hierarchical layers produces a fast, robust and accurate SNS-CF algorithm for visual object tracking. We believe this is going to open a room for improvement about such a fusion. Extensive experimental results on large datasets include LaSOT [21], VOT-2018 [9] and OTB-2015 [10], and shows the effectiveness of SNS-CF algorithm by achieving state-of-the-art results.

Author Contributions: Methodology, Resources, Software and Writing—original draft, T.N.; Investigation, Resources and Validation, H.P. and J.S.; Project administration, Resources and Writing—review & editing J.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government(MSIT) (2017-0-00250, Intelligent Defense Boundary Surveillance Technology Using Collaborative Reinforced Learning of Embedded Edge Camera and Image Analysis).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, Z.; Peng, H. Deeper and wider siamese networks for real-time visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 4591–4600.
2. Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 4282–4291.
3. Ma, C.; Huang, J.B.; Yang, X.; Yang, M.H. Hierarchical convolutional features for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3074–3082.
4. Bertsimas, D.; Dunn, J.; Pawlowski, C.; Zhuo, Y.D. Robust classification. *INFORMS J. Optim.* **2018**, *1*, 2–34. [[CrossRef](#)]
5. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of the European Conference on computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 850–865.
6. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. High performance visual tracking with siamese region proposal network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8971–8980.
7. Thierry, N.; Park, H.; Kim, Y.; Paik, J. SSReF-Spatial_Semantic Residual Features for Object Tracking. *Inst. Electron. Inf. Eng.* **2019**, *11*, 651–654.
8. Wang, Q.; Zhang, L.; Bertinetto, L.; Hu, W.; Torr, P.H. Fast online object tracking and segmentation: A unifying approach. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 1328–1338.
9. Moudgil, A.; Gandhi, V. Long-term visual object tracking benchmark. In Proceedings of the Asian Conference on Computer Vision, Perth, Australia, 2–6 December 2018; Springer: Cham, Switzerland, 2018; pp. 629–645.
10. Wu, Y.; Lim, J.; Yang, M.H. Object tracking benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1834–1848. [[CrossRef](#)] [[PubMed](#)]
11. Gray, R.M. *Toeplitz and Circulant Matrices: A Review*; Now Publishers Inc.: Hanover, MA, USA, 2006.
12. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 583–596. [[CrossRef](#)] [[PubMed](#)]
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
14. Gundogdu, E.; Alatan, A. Good features to correlate for visual tracking. *IEEE Trans. Image Process.* **2018**, *27*, 2526–2540. [[CrossRef](#)] [[PubMed](#)]
15. Shen, J.; Tang, X.; Dong, X.; Shao, L. Visual object tracking by hierarchical attention siamese network. *IEEE Trans. Cybern.* **2019**, *50*, 3068–3080. [[CrossRef](#)] [[PubMed](#)]
16. Hariharan, B.; Arbeláez, P.; Girshick, R.; Malik, J. Hypercolumns for object segmentation and fine-grained localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 447–456.
17. Danelljan, M.; Bhat, G.; Shahbaz Khan, F.; Felsberg, M. Eco: Efficient convolution operators for tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6638–6646.

18. Li, F.; Tian, C.; Zuo, W.; Zhang, L.; Yang, M.H. Learning spatial-temporal regularized correlation filters for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4904–4913.
19. Van Rossum, G.; Drake, F.L. *Python 3 Reference Manual*; PythonLabs: Scotts Valley, CA, USA, 2009.
20. MATLAB. *version 9.8.0.1417392 (R2020a) Update 4*; MATLAB: Natick, MA, USA, 2020.
21. Fan, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Bai, H.; Xu, Y.; Liao, C.; Ling, H. Lasot: A high-quality benchmark for large-scale single object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 5374–5383.
22. Moore, G.; Noyuce, R.O. (Intel Corporation, Santa Clara, CA, USA). Personal communication, 1968.
23. Huang, J.; Priem, C.; Malachowsky, C. (NVIDIA, Santa Clara, CA, USA). Personal communication, 1993.
24. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [[CrossRef](#)]
25. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; Springer: Cham, Switzerland, 2014; pp. 740–755.
26. Xu, N.; Yang, L.; Fan, Y.; Yang, J.; Yue, D.; Liang, Y.; Price, B.; Cohen, S.; Huang, T. Youtube-vos: Sequence-to-sequence video object segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 585–601.
27. Zou, W.; Zhu, S.; Yu, K.; Ng, A.Y. Deep learning of invariant features via simulated fixations in video. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, Harrahs and Harveys, NV, USA, 3–8 December 2012; pp. 3202–3211.
28. Zhang, K.; Zhang, L.; Liu, Q.; Zhang, D.; Yang, M.H. Fast visual tracking via dense spatio-temporal context learning. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; Springer: Cham, Switzerland, 2014; pp. 127–141.
29. Hare, S.; Golodetz, S.; Saffari, A.; Vineet, V.; Cheng, M.M.; Hicks, S.L.; Torr, P.H. Struck: Structured output tracking with kernels. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 2096–2109. [[CrossRef](#)] [[PubMed](#)]
30. Zhong, W.; Lu, H.; Yang, M.H. Robust object tracking via sparse collaborative appearance model. *IEEE Trans. Image Process.* **2014**, *23*, 2356–2368. [[CrossRef](#)] [[PubMed](#)]
31. Zhang, K.; Zhang, L.; Yang, M.H. Fast compressive tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 2002–2015. [[CrossRef](#)] [[PubMed](#)]
32. Babenko, B.; Yang, M.H.; Belongie, S. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *33*, 1619–1632. [[CrossRef](#)] [[PubMed](#)]
33. Kalal, Z.; Mikolajczyk, K.; Matas, J. Tracking-learning-detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *34*, 1409–1422. [[CrossRef](#)] [[PubMed](#)]
34. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. Exploiting the circulant structure of tracking-by-detection with kernels. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 702–715.
35. Sauer, A.; Aljalbout, E.; Haddadin, S. Tracking Holistic Object Representations. *arXiv* **2019**, arXiv:1907.12920.
36. Sun C, Wang D, Lu H, Yang MH. Learning spatial-aware regressions for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8962–8970.
37. Zhu, Z.; Wang, Q.; Li, B.; Wu, W.; Yan, J.; Hu, W. Distractor-aware siamese networks for visual object tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 101–117.
38. Xu, T.; Feng, Z.H.; Wu, X.J.; Kittler, J. Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking. *IEEE Trans. Image Process.* **2019**, *28*, 5596–5609. [[CrossRef](#)] [[PubMed](#)]
39. Bai, S.; He, Z.; Dong, Y.; Bai, H. Multi-hierarchical independent correlation filters for visual tracking. In Proceedings of the 2020 IEEE International Conference on Multimedia and Expo (ICME), London, UK, 6–10 July 2020; pp. 1–6.

40. Bhat, G.; Johnander, J.; Danelljan, M.; Shahbaz Khan, F.; Felsberg, M. Unveiling the power of deep tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 483–498.
41. He, A.; Luo, C.; Tian, X.; Zeng, W. A twofold siamese network for real-time object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 4834–4843.
42. Sun, C.; Wang, D.; Lu, H.; Yang, M.H. Correlation tracking via joint discrimination and reliability learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 489–497.
43. Nam, H.; Han, B. Learning multi-domain convolutional neural networks for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4293–4302.
44. Song, Y.; Ma, C.; Wu, X.; Gong, L.; Bao, L.; Zuo, W.; Shen, C.; Lau, R.W.; Yang, M.H. Vital: Visual tracking via adversarial learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8990–8999.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).